



## تمرین برنامه‌نویسی اول<sup>۱</sup> شبکه‌های کامپیوتری

بهار ۱۳۹۰

مدرس: مهدی خرازی

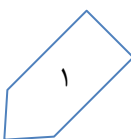
در این تمرین شما ضمن آشنایی با سیستم پرتو، برنامه‌ای را که در تمرین قبلی نوشتید، اندکی ارتقا خواهید داد.

### مقدمه

در بسیاری از شرایط پیش می‌آید که IP ای که به دستگاه‌های مختلف داده می‌شود، IP مجازی نبوده و نمی‌توان در سرتاسر اینترنت توسط آن IP، دستگاه مشخص شده را آدرس دهی کرد و تنها آن آدرس IP، در شبکه‌ی آن دستگاه قابل آدرس‌دهی است. برای اینکه این دستگاه بتواند با شبکه‌های بیرون از خود ارتباط برقرار بکند باید با یک آدرس IP مجاز این ارتباط فراهم شود. به این منظور لازم است که مکانیزمی برای تغییر این آدرس مجازی به یک آدرس مجاز فراهم شود. معمولاً در عمل، این مکانیزم به این صورت پیاده‌سازی می‌شود که کارگزاری در این شبکه وجود دارد که دارای آدرس مجاز می‌باشد. به این کارگزار NAT server گفته می‌شود. دستگاهی که آدرس مجاز ندارد، بسته‌ای را که می‌خواهد به بیرون انتقال دهد را از طریق NAT server منتقل می‌کند و NAT server با تغییر آدرس IP مبدا به آدرس IP خود (که یک آدرس مجاز است)، بسته را انتقال می‌دهد.

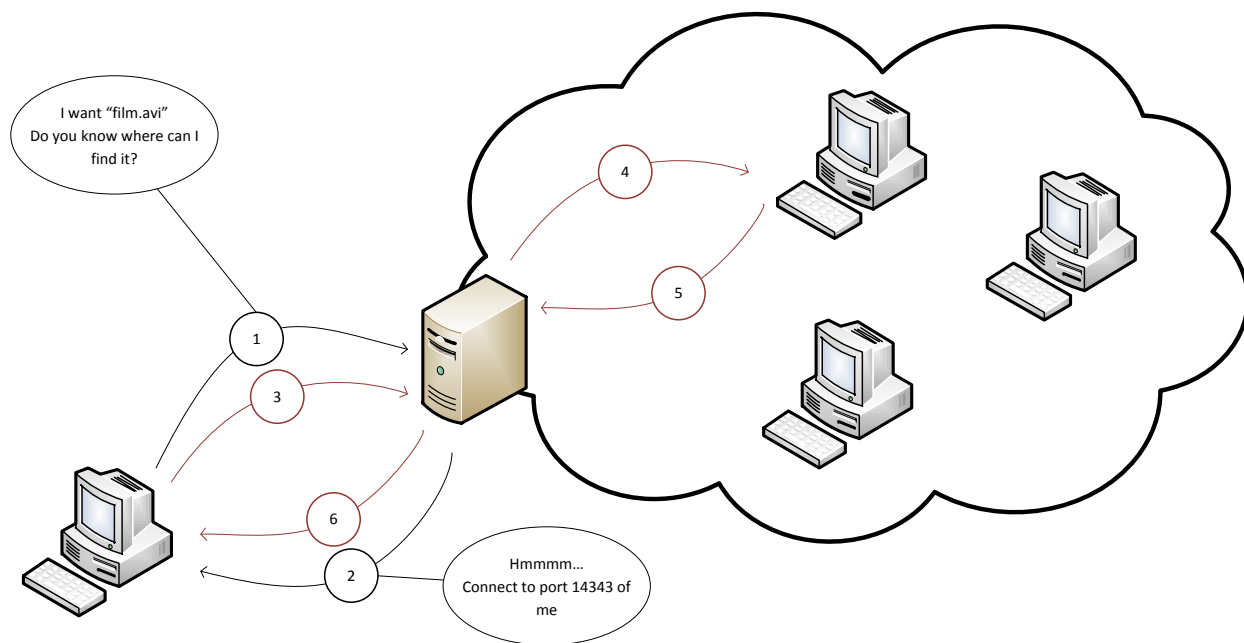
اما نکته‌ی مهمی که در اینجا وجود دارد این است که NAT server، باید کار مشابهی را برای تمامی دستگاه‌های درون آن شبکه انجام دهد. به علاوه چنین کاری را باید برای portهای مختلف هر دستگاه انجام دهد. در صورتیکه به ازای همه‌ی دستگاه‌ها و برای تمامی port آنها، تنها یک port را اختصاص دهد، تمامی بسته‌های ارسال شده، دارای یک آدرس IP و port خواهند بود و موقعی که پاسخی را دریافت می‌کند، نمی‌تواند متوجه شود که مربوط به کدام دستگاه و کدام portش بوده‌است. بنابراین نمی‌تواند بسته‌ی مربوط به پاسخ را به دستگاهی که درخواست را فرستاده بوده، انتقال دهد. به منظور رفع این مشکل، NAT server به ازای هر زوج مرتب IP و port، یک port از خود را اختصاص می‌دهد. بنابراین موقعی که دستگاهی با IP مجازی، بسته‌ای را ارسال

<sup>۱</sup> با تشکر از بهنام مؤمنی، حسن اسلامی، امیر شیخها، علی فتاح‌المنان، فرزانه مقدم، اشکان نیک‌روش و کوشا میرحسینی



می‌کند، NAT server، مطابق با IP و port مربوط به فرستنده آن بسته، یک port از خود را به آن اختصاص می‌دهد و IP فرستنده را به IP واقعی خود و port فرستنده را به portی که به آن زوج مرتب IP و port فرستنده اختصاص داده شده است، تغییر می‌دهد. در صورتیکه NAT server، بسته‌ای را از شبکه اینترنت بگیرد، مشاهده می‌کند که آیا port گیرنده آن بسته، در لیست portهایی که قبلاً به یک زوج مرتب از IP و port اختصاص داده بوده قرار دارد یا خیر. در صورتیکه قرار داشت، باید IP و port گیرنده را به آن زوج مرتب تغییر بدهد. به این صورت یک دستگاه با آدرس مجازی توانایی ارتباط با اینترنت را پیدا کرده است. دقت داشته باشید که شما در این تمرین، NAT را بصورت کامل پیاده‌سازی نمی‌کنید، بلکه Reverse NAT یا Static NAT را پیاده‌سازی خواهید کرد.

در این تمرین لازم است که شما برنامه‌ای را که در تمرین سری قبل نوشتید، اندکی ارتقا دهید. به این صورت که مانند تمرین سری قبل، کاربری دارید که می‌خواهد فایلی را دریافت کند. ولی محل قرار گرفتن این فایل بر روی یک سر host با آدرس IP مجازی است. بنابراین به منظور برقراری ارتباط با این hostها باید از طریق کارگزاری که درون شبکه آن‌ها قرار دارد و دارای IP واقعی است اقدام کرد. بنابراین کاربر برای دریافت یک فایل مشخص، نام آن فایل را به کارگزار ارسال می‌کند. این کارگزار اطلاعات تمامی فایل‌های موجود روی hostها را در اختیار دارد. در صورتیکه مانند تمرین سری قبل، این hostها دارای آدرس واقعی باشند، باید IP و port مربوط به آن host را به عنوان جواب برگرداند. ولی در اینجا، زوج مرتب مربوط به این IP و port را به یک port نگاشت می‌دهد و این port را به عنوان جواب برمی‌گرداند. حال کاربر برای دریافت فایل، مجدداً از طریق پروتکل HTTP، ولی این دفعه بجای فرستادن به IP و port مربوط به آن host، به IP مربوط به کارگزار و port نگاشت داده شده که توسط کارگزار ارسال شده بود، درخواست را می‌فرستد. شکل ۱، روند انجام این کار را نشان می‌دهد.



شکل ۱

## محیط

جهت آشنایی با سیستم پرتو لازم است که برنامه خود را بر روی «چارچوب کاربر» این سیستم پیاده کنید. پرتو این امکان را به شما می‌دهد که تعدادی گره مجازی در شبکه داشته باشید و آن‌ها را مطابق میل خود برنامه‌ریزی کنید. برای آشنایی بیشتر توصیه می‌شود که به مستند «[راهنمای چارچوب کاربر](#)» مراجعه فرمایید.

## انتظارات

شما باید برنامه‌ی مربوط به کاربر و کارگزار را بصورت جداگانه بنویسید. برنامه‌ی مربوط به کاربر، همانند تمرین سری قبل، با استفاده از Socket Programming انجام می‌شود. به این منظور باید فایلی تحت نام `client.cpp` داشته‌باشید. اما برای برنامه‌ی مربوط به کارگزار، به دلیل اینکه در لایه‌های ۳ و ۴ باید تغییر ایجاد شود، بنابراین نیازمند این خواهد بود که بر روی پرتو نوشته‌شود. برنامه‌ی مربوط به کارگزار را باید داخل فایلی تحت نام `sm.cpp` بنویسید. با اجرای فرمان `make`، این دو برنامه کامپایل می‌شوند و دو فایل اجرایی به نام‌های `client.out` و `natserver.out` ایجاد می‌شود. نحوه‌ی اجرا کردن برنامه‌ی مربوط به کاربر و پارامترهای آن همانند تمرین سری قبل است با این تفاوت که تحت دو آرگومان جدید، نام `Host` و آدرس `IP` مبدا را نیز دریافت می‌کند. نحوه‌ی اجرای برنامه‌ی مربوط به کارگزار با پارامترهای مناسب در فایل‌هایی به همراه «چارچوب کاربر» به شما داده شده‌است. در ادامه توضیحات دقیق‌تری در مورد نحوه‌ی کار این دو برنامه داده خواهد شد.

## برنامه‌ها

### برنامه‌ی مربوط به کاربر

برنامه‌ی مربوط به کاربر، تغییر چندانی نسبت به برنامه‌ی کاربری که در تمرین سری قبل نوشته‌اید ندارد. مجدداً برنامه‌ی کاربر، آدرس کارگزار (بصورت `IP` و `Port`)، نام فایل و علاوه بر آنها نام `Host` و آدرس `IP`ی که باید از طریق آن اطلاعات را ارسال کند، بعنوان ورودی دریافت می‌کند. در واقع برای برقراری هر اتصال (چه اتصال `UDP` و چه اتصال `TCP`) باید عملیات `bind` را برای آن آدرس انجام دهد. سپس با برقراری ارتباطی از نوع `UDP` با کارگزار، نام فایل را به کارگزار می‌فرستد (بصورت رشته‌ای از کاراکترها که به `0` ختم می‌شود). در اینجا کارگزار `Port`ی از خود را به عنوان پاسخ برمی‌گرداند (توجه کنید که این داده باید بصورت `Network Byte Order` فرستاده شود). مانند تمرین سری قبل، در صورتیکه کارگزار تا ۱ ثانیه پاسخ نداد، کاربر دوباره باید آدرس فایل مورد نظرش را از کارگزار بپرسد. حال کاربر باید درخواست `HTTP GET` را به `Port` مشخص شده از کارگزار تحت ارتباطی از نوع `TCP` بفرستد. به این صورت توانسته فایلی را از `host`ی که `IP` مجازی دارد، دریافت کند.

Filename
"/index.html\0"

شکل ۲ - داده‌ی مربوط به درخواست فایل توسط `UDP`

Port (2 bytes)
1391

شکل ۳- داده‌ی مربوط به پاسخ درخواست UDP

## برنامه‌ی مربوط به کارگزار

بطور کلی برنامه‌ی مربوط به کارگزار باید دو کار اساسی انجام دهد. یکی اینکه پاسخ درخواست‌های کاربر مبنی بر آدرس قرارگیری یک فایل را بدهد. و دیگر اینکه بسته‌هایی را که از یک کاربر به یک host یا برعکس فرستاده می‌شود، انتقال دهد.

در صورتیکه بسته‌ای که به کارگزار می‌رسد از نوع UDP باشد، مشخص می‌شود که از نوع اول است. بنابراین این بسته، نام یک فایل را دربردارد. کارگزار باید مشخص کند که IP و Port مربوط به hostی که این فایل را دارد، چیست. ولی همانگونه که توضیح داده‌شد، باید این زوج مرتب را به Port آزادی از خود نگاشت کند و در جواب آدرس Port نگاشت‌شده را بفرستد. به این منظور داده‌ای که فرستاده می‌شود باید حاوی ۲ بایت باشد که همان Port نگاشت‌شده می‌باشد. در صورتیکه کارگزار آدرس محل قرارگیری فایل را نداند، این ۲ بایت را صفر قرار خواهد داد.

در صورتیکه بسته‌ای که به کارگزار می‌رسد، از نوع TCP باشد، مشخص می‌شود که از نوع دوم است. به این معنا که باید این بسته را با ایجاد تغییراتی در لایه‌ی IP و TCP (و حتی Ethernet!) انتقال دهد. در صورتیکه بسته از سمت کاربر آمده‌باشد، باید IP و Port گیرنده را به IP و Port متعلق به host مربوطه، تغییر دهد. این کار از طریق جدولی که حاوی نگاشت مربوط به IP و Port متعلق به هر host، به Portهای آزاد کارگزار است، انجام می‌شود. در صورتیکه بسته از سمت host آمده‌باشد، مجدداً باید کار مشابهی را روی IP و Port فرستنده انجام دهد. ولی این دفعه باید IP و Port فرستنده را با استفاده از جدول ذکر شده، به IP خود و Port نگاشت‌شده تغییر دهد.

Port Number for UDP transfer

MAC Address of Gateway

Number of Hosts

<IP> }  
<MAC> } For each host

<File> }  
<IP> } For each file  
<Port> }

## نکات ضروری

- برای آشنایی با سرایند لایه‌های به کار رفته در این تمرین، به لینک‌های زیر مراجعه کنید:
  - <http://www.security-freak.net/raw-sockets/raw-sockets.html>
  - <http://en.wikipedia.org/wiki/IPv4>
  - [http://en.wikipedia.org/wiki/Transmission\\_Control\\_Protocol](http://en.wikipedia.org/wiki/Transmission_Control_Protocol)
  - [http://en.wikipedia.org/wiki/User\\_Datagram\\_Protocol](http://en.wikipedia.org/wiki/User_Datagram_Protocol)
- در صورتیکه هر مشکل یا پرسشی داشتید که فکر می‌کنید پاسخ آن برای همه مفید خواهد بود، لطفاً آن را به گروه پستی درس ارسال کنید.
- از فرستادن جواب تمرین به گروه پستی جداً خودداری کنید.
- فرستادن کل یا قسمتی از برنامه‌تان برای افراد دیگر، یا استفاده از کل یا قسمتی از برنامه فردی دیگر به نام خود، تقلب محسوب می‌شود.
- پس از اتمام کارتان لازم است که پوشه user را به همراه Makefile فشرده کرده و بر روی سیستم خودکار داوری<sup>۲</sup> upload کنید.

موفق باشید

<sup>2</sup> <http://partov.sharif.edu/judge>